Important Note | Coursera

coursera.org/learn/python-aws-elastic-beanstalk-web-application/supplement/ujVS2/important-note

Important Note

To avoid distraction for set up during the course, we would recommend that you create an Amazon AWS account beforehand. Amazon AWS provides a free tier option for 1 year & the course materials will utilize services that fall under the free tier option. Create your free Amazon Web Services (AWS) account by going to https://aws.amazon.com/free

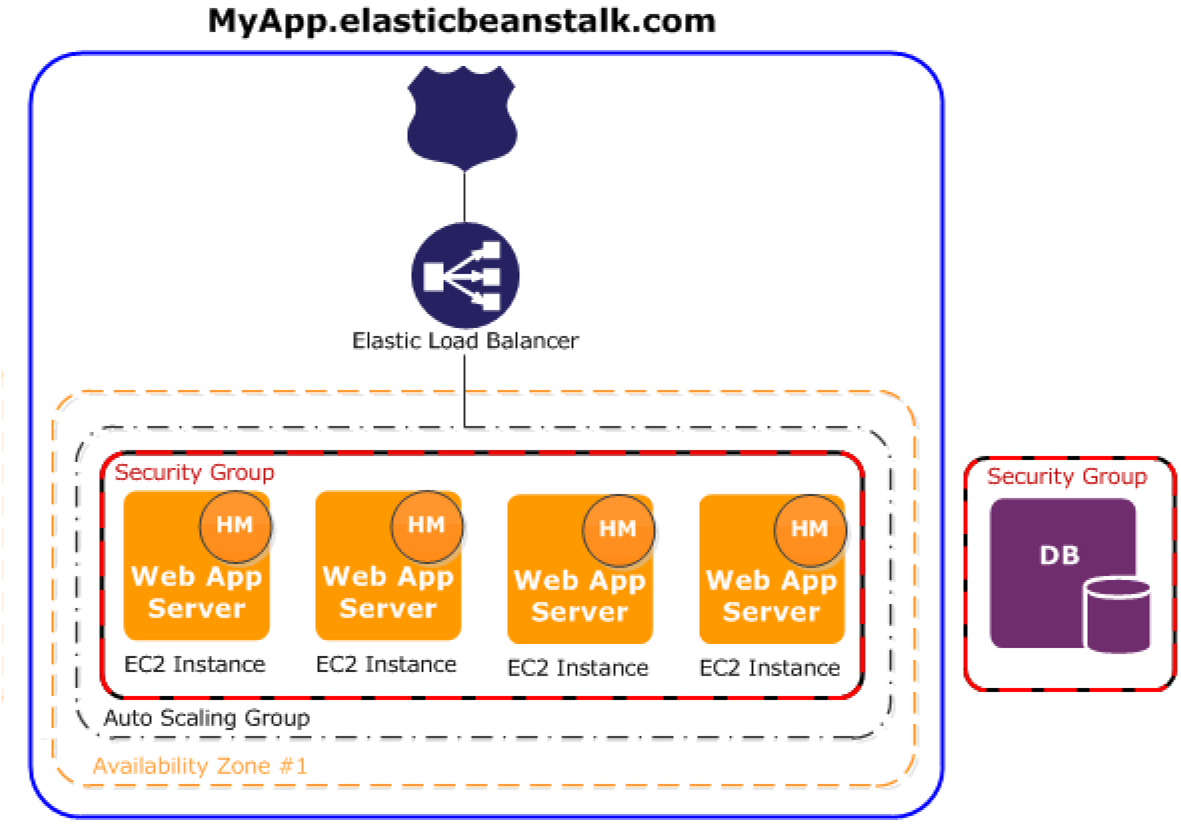
Vocabulary

Please review the following vocabulary items before taking the project:

- **Flask** is a micro web framework written in Python. It is classified as a microframework because of its minimal structure that provides an ecosystem of external components such as form validation, upload handling, authentications, etc., that can be mixed and matched as needed, focusing on including only what you need instead of providing a complete toolset. Companies that use the Flask framework include Pinterest and LinkedIn. [Source: Wikipedia]

- **Amazon Web Services (AWS)** is a subsidiary of Amazon providing on-demand cloud computing platforms and APIs to individuals, companies, and governments, on a metered pay-as-you-go basis. These cloud computing web services provide a variety of basic abstract technical infrastructure and distributed computing building blocks and tools. [Source: AWS]

  Amazon Elastic Compute Cloud (Amazon EC2), which provides secure, resizable compute capacity in the cloud. In simple terms cloud servers. [Source: amazonaws.cn/en/ec2/]



The environment is the heart of the application. In the diagram, the environment is shown within the top-level solid line. When you create an environment, Elastic Beanstalk provisions the resources required to run your application. AWS resources created for an environment include one elastic load balancer (ELB in the diagram), an Auto Scaling group, and one or more Amazon Elastic Compute Cloud (Amazon EC2) instances.

Every environment has a CNAME (URL) that points to a load balancer. The environment has a URL, such as myapp.us-west-2.elasticbeanstalk.com. This URL is aliased in Amazon Route 53 to an Elastic Load Balancing URL—something like abcdef-123456.us-west-2.elb.amazonaws.com—by using a CNAME record. Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. It provides secure and reliable routing to your infrastructure. The domain name that you registered with your DNS provider will forward requests to the CNAME.

The load balancer sits in front of the Amazon EC2 instances, which are part of an Auto Scaling group. Amazon EC2 Auto Scaling automatically starts additional Amazon EC2 instances to accommodate increasing load on your application. If the load on your application decreases, Amazon EC2 Auto Scaling stops instances, but always leaves at least one instance running.

The Amazon EC2 instances shown in the diagram are part of one security group. A security group defines the firewall rules for your instances. By default, Elastic Beanstalk defines a security group, which allows everyone to connect using port 80 (HTTP). You can define more than one security group. For example, you can define a security group for your database server.

> **Environment properties** are a set of name/value pair variables whose value is set outside the program, typically through a functionality built into the operating system or microservice. These are loaded during application initialization. [Source: Jim Medlock]